AGILENT TECHNOLOGIES
Legal Department, M/S DL429
Intellectual Property Administration
P.O. Box 7599
Loveland, CO 80537-0599

PATENT APPLICATION
ATTORNEY DOCKET NO.10991663-1

IN THE
UNITED STATES PATENT AND TRADEMARK OFFICE

INVENTOR(S): Que-Won Rhee **CONFIRMATION NO:** 1594

SERIAL NO: 09/432,819 **GROUP ART UNIT:** 2182

FILED: November 2, 1999 **EXAMINER:** Ilwoo Park

SUBJECT: CONFIGURABLE ARCHITECTURE FOR VIRTUAL SOCKET CLIENT TO AN ON-CHIP BUS INTERFACE BLOCK

ASSISTANT COMMISSIONER OF PATENTS
P.O. Box 1450
Alexandria, VA 22313-1450

SIR:

APPEAL BRIEF

Appellant herein sets forth his reasons and arguments for appealing the Examiner's final rejection of claims in the above-identified case.

REAL PARTY IN INTEREST

This Patent Application has been assigned to Agilent Technologies, Inc., which has been incorporated in the State of Delaware.

RELATED APPEALS AND INTERFERENCES

Appellant is aware of no related appeals or interferences.

STATUS OF CLAIMS

Claims 1 through 12 are extant in the case.

Claims 1, 2, 4 and 6 through 12 are rejected.

Claims 3 and 5 are objected to.

The appealed claims are claims 1, 2, 4 and 6 through 12.

STATUS OF AMENDMENTS

After the final rejection, Appellant filed a Response to Office Action dated March 14, 2003. In the Response to Office Action, no amendments were made to the claims.

SUMMARY OF THE INVENTION

Within an integrated circuit, it is sometimes necessary for a port of a specialized logic block to interface with an on-chip bus. For example the specialized logic block is proprietary to a particular vendor. The interface can be facilitated by modifying the specialized logic block so that the port of the logic block will interface directly with the on-chip bus; however, modifying a specialized logic block may introduce errors and requires extensive internal knowledge and re-verification time. See the Specification at page 1, lines 8 through 16.

Alternatively, a special interface block can be introduced that provides an interface between the port of the logic block and the on-chip bus; however, design of such a special interface block is difficult and time

consuming. Further, any variation in the configuration requirements of the interface can require a complete redesign of the interface block. See the Specification at page 1, lines 11 through 14.

In the preferred embodiment of the present invention, a standardized interface block (19) is provided that is modular, facilitating rapid, system-on-chip implementations.

Specifically, an interface block (19) provides an interface between an internal bus (15) of an integrated circuit (200) and a socket (20,25) of a logic block (10). See Figure 1 and the Specification at page 3, lines 12 through 15. The interface block (19) includes a synchronization module (11) that performs any needed synchronization between a clock domain of the internal bus (15) and a clock domain of the socket (20,25) of the logic block (10). A translation module (12) provides translation of block encoding of the data for data transferred between the internal bus (15) and the socket (20,25) of the logic block (10). A queue module (13) buffers data flowing between the internal bus (15) and the socket (20,25) of the logic block (10). A driver module (14) handles low level and electrical drive specifications of the internal bus (15). See Figure 2 and the Specification at page 4, line 3 through page 5, line 9.

Each of the modules can be individually customized as needed. For example, the synchronization module (11) can be implemented as a null synchronization block (61) where no synchronization is required between clock domains (See Figure 4), as a ratio synchronization block (81) where the

clock domain of the internal bus (15) is related to the clock domain of the socket (20,25) of the logic block (10) by a fixed multiplier ratio (See Figure 5), or as a full synchronization block (101) where there is no phase relationship between the clock domain of the internal bus (15) and the clock domain of the socket (20,25) of the logic block (10) (See Figure 6). Also, see the Specification at page 7, lines 1 through 15.

ISSUES PRESENTED FOR REVIEW

The following issues are presented for review:

(1) whether under 35 U.S.C. § 102 (e) claims 1, 2, 4, and 6 through 12 are anticipated by USPN 5,870,310 (*Malladi*).

GROUPING OF CLAIMS

The rejected claims do not stand or fall together. The claims 1, 2, 4, and 6 through 12 are divided into four groups. The first group contains claim 1. The second group contains claim 2. The third group contains claims 4 and 6. The fourth group contains claims 7 through 12.

ARGUMENT

A. Overview Specifying Errors in the Rejection of the Claims.

The criteria for a rejection under 35 U.S.C. § 102 has been clearly defined by the courts and confirmed by the U.S. Patent and Trademark Office. "A claim is anticipated only if each and every element as set forth in the claim is found, either expressly or inherently described, in a single prior art

reference." *Verdegaal Bros. v. Union Oil Co. of California*, 814 F.2d 628, 631, 2 USPQ2d 1051, 1053 (Fed. Cir. 1987). "The identical invention must be shown in as complete detail as is contained in the ... claim." *Richardson v. Suzuki Motor Co.*, 868 F.2d 1226, 1236, 9 USPQ2d 1913, 1920 (Fed. Cir. 1989).

Examiner has failed to show that each and every element set forth in the claims is found either expressly or inherently in Malladi. Below, Applicant clearly and unambiguously points out subject matter within each independent claim that is not disclosed by Malladi.

B. Description of *Malladi.*

Malladi discloses a method and apparatus for designing re-usable core interface shells. The interface shells provide interface functions between a hardware core and one or more busses. The hardware circuitry of the shells includes circuitry for bus interface units, memory interface units, buffers, and bus protocol logic. See the Abstract.

C. Discussion of Group 1 claim (claim 1).

1. Subject matter within claim 1 not disclosed by Malladi

Claim 1 sets out an interface block that provides an interface between an internal bus and a socket of a logic block. The interface block comprises four separate modules: a synchronization module, a translation module, a queue module and a driver module.

Malladi does not disclose or suggest an interface between an internal bus and a socket of a logic block, where the interface comprises a

synchronization module, a translation module, a queue module and a driver module.

Malladi does disclose interface logic cells that can be used as an interface between an internal bus and a logic block. However, Malladi does not provide implementation details for any of the interface logic cells. Nothing in Malladi discloses or suggests that the interface logic cells are modular. Nothing in Malladi discloses or suggests the particular modules (a synchronization module, a translation module, a queue module and a driver module) set out by claim 1.

For example, Figure 1 of Malladi shows an interface logic cell 116a used to provide an interface between CPU core A 104 and CPU bus 100. However, Malladi gives no details of how interface logic cell 116a is implemented, except to state that interface logic cell 116a may include complex interface logic (e.g., BUI's, staging buffers, bus protocols) that has been previously designed, characterized, tested and ascertained to be dependable. See column 4, lines 4 through 9.

Since Malladi does not give any indication that the disclosed interface logic cells are modular and does not disclose or suggest the particular modules (a synchronization module, a translation module, a queue module and a driver module) set out by claim 1, it is clear Malladi does not anticipate claim 1 under 35 U.S.C. § 102.

2. Errors made by the Examiner in the stated rationale for the Rejection

The Examiner has made incorrect correlations between the subject matter of the present invention and entities within Malladi. Essentially, the Examiner has tried to treat interface logic cells in Malladi as if they were "modules" within an interface block within an interface block between an internal bus and a socket of a logic block. The trouble with this is that in Malladi, the interface between a logic block and an internal bus is in every case implemented by only a single interface block. Thus, no two interface blocks in Malladi provide an interface between the same two entities. Each interface block in Malladi is a complete interface between two entities and cannot be considered "modules" within an interface block between an internal bus and a socket of a logic block.

For example, claim 1 of the present case states that an interface block between an internal bus and a socket of a logic block includes a synchronization module and a driver module. However, the examples that the Examiner gives from Malladi of a "synchronization module" (e.g., MIUs 150a-150c) and a "driver module" (e.g., BIUs 140a-140c) are not within the same logic block *and do not even interface to the same bus!* That is, MIUs 150a-150c interface to memory bus 102 and BIUs 140a-140c interface to CPU bus 100.

More specifically, the Examiner asserts that memory interface units (MIUs) 150a-150c are examples of translation modules set out in claim 1 of the present invention. See the Office Action dated February 4, 2003 at page 3, lines 7 and 8. However, as can be seen from Figure 1 of Malladi, MIU 150a

provides a complete interface between data processing core 108 and memory bus 102. MIU 150b provides a complete interface between data processing core 110 and memory bus 102. MIU 150c provides a complete interface between data processing core 112 and memory bus 102.

At the same time, the Examiner asserts that bus interface units (BIUs) 140a-140c are examples of driver modules set out in claim 1 of the present invention. See the Office Action dated February 4, 2003 at page 3, lines 17 and 18. However, as can be seen from Figure 1 of Malladi, BIU 140a provides a complete interface between data processing core 108 and CPU bus 100. BIU 140b provides a complete interface between data processing core 110 and CPU bus 100. BIU 150c provides a complete interface between data processing core 112 and CPU bus 100.

Thus, as can be seen from Figure 1, the examples that the Examiner gives from Malladi of a "synchronization module" (MIUs 150a-150c) and a "driver module" (BIUs 140a-140c) are not within the same logic block and do not interface to the same bus.

As discussed above, the entities with Malladi that most closely correspond to the interface block set out in claim 1 are the interface logic cells. However, Malladi does not provide implementation details for any of the interface logic cells. Nothing in Malladi discloses or suggests that the interface logic cells are modular. Nothing in Malladi discloses or suggests the particular modules (a synchronization module, a translation module, a queue module and a driver module) set out by claim 1.

D. Discussion of Group 2 claim (claim 2).

1. Subject matter within claim 2 not disclosed by Malladi

Claim 2 sets out an interface block as in claim 1 wherein the synchronization module can be implemented as a null synchronization block, a ratio synchronization block, or a full synchronization block.

This is not disclosed or suggested by Malladi. Malladi does not disclose a synchronization module and does not disclose a synchronization module being implemented as a null synchronization block, a ratio synchronization block, or a full synchronization block.

2. Errors made by the Examiner in the stated rationale for the Rejection

The Examiner merely states without further explanation that Malladi teaches a synchronization module that can be implemented as a null synchronization block or a ratio synchronization block. See the Office Action dated February 4, 2003 at page 3, line 19 through page 4, line 4. However, there appears to be nothing in Malladi that teaches a synchronization module that can be implemented as a null synchronization block or a ratio synchronization block

The Examiner cites Malladi at column 3, lines 42 through 48 and Column 4, lines 43 through 56 as teaching a full synchronization block where there is no phase relationship between the clock domain of the internal bus and the clock domain of the socket of the logic block. However, neither in

these sections nor anywhere else in Malladi is there any discussion of synchronization between the clock domain of an internal bus and the clock domain of the socket of a logic block, as set out in claim 2 of the present application.

Since Malladi is silent about all the limitations set out in claim 2, it is clear that Malladi does not disclose or suggest the subject matter of claim 2.

E. Discussion of Group 3 claims (claims 4 and 6).

1. Subject matter within claim 4 not disclosed by Malladi

Claim 4 sets out a method for providing an interface between an internal bus of an integrated circuit and a socket of a logic block within the integrated circuit.

In each of the steps of the method set out in claim 4, specific modules perform specific functions. In step (a), a synchronization module performs any needed synchronization between a clock domain of the internal bus and a clock domain of the socket of the logic block. As discussed above, no such synchronization module is disclosed by Malladi.

Examiner has asserted a synchronization module is disclosed by memory interface units 150a, 150b, 150c, 250b, 312, 326, or 336. However , memory interface units 150a, 150b, 150c, 250b, 312, and 326 are memory interface units provide complete bus interface between a data processing core (108, 110, 112, etc.) and the memory bus (102, 222, 314). Nowhere does Malladi indicate that memory interface units 150a, 150b, 150c, 250b, 312, 326,

or 336 perform synchronization between a clock domain of an internal bus and a clock domain of the socket of the logic block, as set out in step (a) of claim 4. Malladi is essentially silent as to whether this functionality is performed by the MIUs.

However, any interface functionality that needs to be performed between, for example, memory bus 102 and data processing core 108, is handled by memory interface unit 150a. Memory interface units 150a, 150b, 150c, 250b, 312, and 326 do not function as synchronization modules but operate as complete memory bus interfaces for their respective logic processing cores.

In step (b), a translation module is used to provide any required translation of block encoding of data transferred between the internal bus and the socket of the logic block. No such translation module is disclosed by Malladi.

Examiner cites Malladi at column 5, lines 2 through 7 as disclosing this functionality. In this section, Malladi indicates that additional interface logic cells may represent staging buffers, local memory blocks, communications protocols etc. However, Malladi does not specifically disclose or suggest in this section (or any other section) a translation module that, for data transferred between an internal bus and a socket of a logic block, provides translation of block encoding of the data.

In step (d), a driver module is used to handle low level and electrical drive specifications of the internal bus. No such driver module is disclosed by Malladi.

Examiner has asserted a driver module is disclosed by bus interface units 140a, 140b, 140c, 240b, 310, 336, or 340. While bus interface units 140a, 140b, 140c, 240b, 310, 336, and 340 may handle low level and electrical drive specifications (Malladi is silent on the issue), these memory interface units are not modules dedicated to performing this functionality, but rather these BIUs provide complete bus interface between a data processing core (108, 110, 112, etc.) and the CPU bus (100, 220, 316). Thus any functionality that needs to be performed between, for example, CPU bus 100 and data processing core 108, is handled by bus interface unit 140a. Bus interface units 140a, 140b, 140c, 240b, 310, 336, or 340 operate as complete memory bus interfaces for their respective logic processing cores.

2. Errors made by the Examiner in the stated rationale for the Rejection

In the rejection of claim 4, the Examiner has tried to treat interface logic cells in Malladi as if they were "modules" within an interface block within an interface block between an internal bus and a socket of a logic block. The trouble with this is that in Malladi, the interface between a logic block and an internal bus is in every case implemented by only a single interface block. Thus, no two interface blocks in Malladi provide an interface between the same two entities. It is clear then that each interface block in Malladi is a complete interface between two entities and cannot be considered "modules" within an interface block between an internal bus and a socket of a logic block.

As discussed above, the entities within Malladi that provide an interface between an internal bus and a logic block are the interface logic cells. However, Malladi does not provide implementation details for any of the interface logic cells. Nothing in Malladi discloses or suggests that the interface logic cells are modular. Nothing in Malladi discloses or suggests the particular modules (a synchronization module, a translation module, a queue module and a driver module) set out by claim 4.

F. Discussion of Group 4 claims (claims 7 through 12).

1. Subject matter within claim 7 not disclosed by Malladi

Independent claim 7 sets out an interface block that provides an interface between an internal bus of the integrated circuit and a socket of a logic block. The interface block comprises a plurality of modules connected in series. This is not disclosed or suggested by Malladi. Specifically, nowhere in Malladi is there disclosed an interface block that comprises a plurality of modules. Nowhere in Malladi is there disclosed a plurality of modules connected in series.

In claim 7, each module in the plurality of modules performs only a single function from a plurality of functions. Any needed synchronization between a clock domain of the internal bus and a clock domain of the socket of the logic block is a first function from the plurality of functions. Any required translation of block encoding of data is a second function from the plurality of functions. Any buffering of data flowing between the internal bus and the socket of the logic block is a third function from the plurality of functions. Any low level and electrical drive specifications of the internal bus

is a fourth function from the plurality of functions. This is not disclosed or suggested by Malladi. Particularly, Malladi does not disclose the specific functions of synchronization, translation, buffering, and low level and drive specifications being performed in a module that performs only one of these functions.

2. Errors made by the Examiner in the stated rationale for the Rejection

As discussed above, claim 7 sets out a plurality of modules connected in series. Each module in the plurality of modules performs only a single function from a plurality of functions. Examiner has argued that logic blocks 302, 304, 306, 308, 310 or 312 are examples of the plurality of modules connected in series. However, as can be seen from Figure 3, logic blocks 302, 304, 306, 308, 310 or 312 are not connected in series.

The Examiner also variously asserts that memory interface units 150a, 150b, 150c, 250b, 312, and 326, and bus interface units 140a, 140b, 140c, 240b, 310, 336, and 340 of Malladi perform the functions of modules in the plurality of modules as set out by claim 7. However, as discussed above, each of these interface units disclosed in Malladi performs a complete interface between two entities (e.g., a logic core and a bus). None of the interface units within Malladi are a module of an interface block that perform only a single function from a plurality of functions, as set out by claim 7.

CONCLUSION

For all the reasons discussed above, Appellant believes the rejection of the claims was in error and respectfully requests that the rejection be reversed.

Respectfully submitted,
QUE-WON RHEE

By _____
Douglas L. Weller
Reg. No. 30,506

June 5, 2003
Santa Clara, California
(408) 985-0642

Appendix: Appealed Claims

1. (Previously Amended) An interface block that provides an interface between an internal bus and a socket of a logic block, the interface block, the internal bus and the logic block all being located within a single integrated circuit, the interface block comprising:

a synchronization module that performs any needed synchronization between a clock domain of the internal bus and a clock domain of the socket of the logic block;

a translation module that, for data transferred between the internal bus and the socket of the logic block, provides translation of block encoding of the data;

a queue module, that buffers data flowing between the internal bus and the socket of the logic block; and,

a driver module that handles low level and electrical drive specifications of the internal bus.

2. An interface block as in claim 1 wherein the synchronization module can be implemented as one of:

a null synchronization block where no synchronization is required between the clock domain of the internal bus and the clock domain of the socket of the logic block;

a ratio synchronization block where the clock domain of the internal bus is related to the clock domain of the socket of the logic block by a fixed multiplier ratio; and,

a full synchronization block where there is no phase relationship between the clock domain of the internal bus and the clock domain of the socket of the logic block.

4. A method for providing an interface between an internal bus of an integrated circuit and a socket of a logic block within the integrated circuit, the method comprising the steps of:

(a) performing any needed synchronization between a clock domain of the internal bus and a clock domain of the socket of the logic block within a synchronization module;

(b) providing any required translation of block encoding of data transferred between the internal bus and the socket of the logic block using a translation module;

(c) buffering data flowing between the internal bus and the socket of the logic block using a queue module; and,

(d) handling low level and electrical drive specifications of the internal bus using a driver module.

6. A method as in claim 4 additionally comprising the following step:

(e) providing buffers between modules to allow pipelined operation.

7. (Previously Amended) On an integrated circuit, an interface block that provides an interface between an internal bus of the integrated circuit and a socket of a logic block, the interface block comprising:

a plurality of modules connected in series, wherein each module in the plurality of modules performs only a single function from a plurality of functions;

wherein any needed synchronization between a clock domain of the internal bus and a clock domain of the socket of the logic block is a first function from the plurality of functions, any required translation of block encoding of data is a second function from the plurality of functions, any buffering of data flowing between the internal bus and the socket of the logic block is a third function from the plurality of functions, and handling any low level and electrical drive specifications of the internal bus is a fourth function from the plurality of functions.

8. An interface block as in claim 7 wherein a first module in the plurality of modules is a synchronization module that performs any needed synchronization between the clock domain of the internal bus and the clock domain of the socket of the logic block.

9. An interface block as in claim 7 wherein one module in the plurality of modules is a translation module that, for the data transferred between the internal bus and the socket of the logic block, provides translation of block encoding of the data.

10. An interface block as in claim 7 wherein one module in the plurality of modules is a queue module, that buffers the data flowing between the internal bus and the socket of the logic block.

11. An interface block as in claim 7 wherein one module in the plurality of modules is a driver module that handles low level and electrical drive specifications of the internal bus.

12. An interface block as in claim 7 additionally comprising a plurality of buffers situated between modules in the plurality of modules, the buffers used to pipeline the interface block.

AGILENT TECHNOLOGIES, INC.
Legal Department, DL429
Intellectual Property Administration
P.O. Box 7599
Loveland, Colorado 80537-0599

AF/2182$

PATENT APPLICATION

ATTORNEY DOCKET NO. 10991663-1

2100

IN THE
UNITED STATES PATENT AND TRADEMARK OFFICE

Inventor(s): Que-Won Rhee

Serial No.: 09/432,819 **Examiner:** Ilwoo Park

Filing Date: 11/2/99 **Group Art Unit:** 2182

Title: CONFIGURABLE ARCHITECTURE FOR VIRTUAL SOCKET CLIENT TO AN ON-CHIP BUS INTERFACE BLOCK

ASSISTANT COMMISSIONER FOR PATENTS
PO Box 1450
Alexandria, VA 22313-1450

TRANSMITTAL OF APPEAL BRIEF

Sir:

Transmitted herewith in triplicate is the Appeal Brief in this application with respect to the Notice of Appeal filed on **April 18, 2003**.

The fee for filing this Appeal Brief is (37 CFR 1.17(c)) $320.00.

(complete (a) or (b) as applicable)

The proceedings herein are for a patent application and the provisions of 37 CFR 1.136(a) apply.

( ) (a) Applicant petitions for an extension of time under 37 CFR 1.136 (fees: 37 CFR 1.17(a)-(d) for the total number of months checked below:

( ) one month	$110.00
( ) two months	$410.00
( ) three months	$930.00
( ) four months	$1450.00

( ) The extension fee has already been filled in this application.

(X) (b) Applicant believes that no extension of term is required. However, this conditional petition is being made to provide for the possibility that applicant has inadvertently overlooked the need for a petition and fee for extension of time.

Please charge to Deposit Account 50-1078 the sum of **$320.00**. At any time during the pendency of this application, please charge any fees required or credit any overpayment to Deposit Account 50-1078 pursuant to 37 CFR 1.25.

(X) A duplicate copy of this transmittal letter is enclosed.

(X) I hereby certify that this correspondence is being deposited with the United States Postal Service as first class mail in an envelope addressed to: Commissioner for Patents, PO Box 1450, Alexandria, VA 222313-1450.
Date of Deposit: **June 6, 2003** or

I hereby certify that this paper is being facsimile transmitted to the Patent and Trademark Office on the date shown below.
( ) Date of Facsimile:

Typed Name: **Douglas L. Weller**
Signature

Respectfully submitted,

Que-Won Rhee

By

Douglas L. Weller
Attorney/Agent for Applicant(s)
Reg. No. **30,506**
Date: **June 5, 2003**

Telephone No.: **(408) 985-0642**